

## EVALUATING OPTIMIZATION METHODOLOGIES FOR FUTURE INTEGRATION IN BUILDING PERFORMANCE TOOLS

Michael T.M. Emmerich<sup>‡</sup>, Christina Hopfe<sup>†</sup>, Robert Marijt<sup>‡</sup>, Jan Hensen<sup>†</sup>, Christian Struck<sup>†</sup>, and Paul Stoelinga<sup>\*</sup>

<sup>†</sup>Building Physics and Systems, Technical University Eindhoven, The Netherlands;

<sup>‡</sup>LIACS, University of Leiden, 2333 CA-Leiden, The Netherlands;

<sup>\*</sup>Deerns consulting engineers, Rijswijk, The Netherlands

Corresponding author: emmerich@liacs.nl

### ABSTRACT

Building performance simulation (BPS) is a powerful tool to predict and analyze the dynamic behavior of indicators such as energy consumption and comfort among others. Previous work has shown that the use of BPS is mostly limited to code compliance checking in the detailed design. Our long-term goal is to improve the use of BPS in the later phases of the design process by, for instance, indicating design solutions and by introducing integrated building and system optimization. In this context, we hypothesize that introducing a design optimization and exploration capability to BPS-tools can provide a valuable support for decision making.

This paper presents preliminary results and experiences on extending an existing BPS – tool with a capability for single-and multiobjective optimization and parameter exploration. The focus of this work is on energy consumption and thermal comfort.

### 1. INTRODUCTION

For the analysis and prediction of the dynamic behavior of building performance indicators such as energy consumption and thermal comfort, building performance simulation (BPS) is a powerful tool. Previous work has shown that the use of BPS is mostly limited to code compliance checking in the detailed design [13].

The use of multi-disciplinary optimization [10] in building design is still in its early stages [5]. Disciplinary specific optimization activities are known by Michalek et al [8] in architectural design and for instance by Wright et al. [14] in mechanical engineering.

In this publication we propose the introduction of a spectrum of optimization and design space exploration techniques that can be used by the engineer to find the optimal solution for a building regarding the performance with respect to the often conflicting objectives.

Considered parameters being optimized are e.g. the percentage of glass, and the infiltration rate among others.

The paper is structured as follows: Section 2 introduces the building performance optimization problem. Section 3 provides a brief description of algorithms studied for this application. Section 4

presents results of Design of Experiments and trial-and-error strategies, while section 5 discusses results achieved with different adaptive optimization techniques. In section 6, we discuss Pareto optimization as an alternative solution method for problems with multiple, conflicting objectives. Finally, in section 7 we summarize results and outline future research.

### 2. PROBLEM DEFINITION

The objective function below estimates the total energy consumption for heating and cooling in a building:

$$P_{\text{verw.koel}} = P_{\text{w.vent}} + P_{\text{w.inf}} + P_{\text{w.verl}} + P_{\text{w.app}} + C$$

$P_{\text{w.vent}}$  stands for the heating consumption to provide natural ventilation in a building.  $P_{\text{w.inf}}$  is the number of air exchanges per hour caused by unwanted air infiltration from the outside.  $P_{\text{w.verl}}$  and  $P_{\text{w.app}}$  respectively denote the effects of the load due to the equipment and load of lighting due to the energy consumption. There are additional terms to be considered, such as the heat transfer through the different boundaries in the room, the effect of solar radiation that directly enters the building through convection and the accumulated warmth by the construction. In this study they are summarized by a constant value  $C$ .

A case study was performed based on a hypothetical building (see figure 1) which is part of an international test method for assessing the accuracy of BPS tools with respect to various building performance parameters [6].

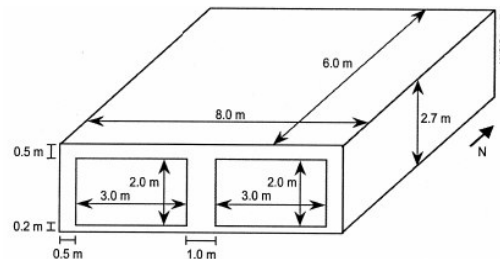


Figure 1. BESTTES Case 600 - Geometry

Lea pre-release v0.9.1 (for beta testing), was chosen as BPS tool. This is a design analysis tool specifically developed for Dutch professionals to predict heating

and cooling peak loads and energy consumption, respectively.

As a simplified measure for thermal comfort the number of hours of air temperatures exceeding 28°C were calculated. A Dutch Standard suggests the use of weighted overheating hours to assess thermal comfort, but, as the tool does not provide a mean radiant surface temperature, air temperature has been used as a simplified measure for thermal comfort.

Due to this simplification the number of hours above 28°C is large. Although air temperature is of limited use when assessing thermal comfort, it has value when comparing the performance of different optimization algorithms.

The following parameters were chosen from the list of building characteristics and will be varied as parameters in the optimization process:

- Infiltration rate: air exchange rate per hour in the building
- Window fraction: the amount of glass percentage on one wall of the building
- Load equipment: power of equipment per net floor surface area [W/m<sup>2</sup>]
- Load lighting: power of lighting per net floor surface area [W/m<sup>2</sup>]

To determine the best algorithm for optimizing the appropriate objective function we study three methodologies for optimization and exploration, namely non-adaptive algorithms, adaptive optimization, and Pareto optimization, will be made. In the next paragraphs these methodologies and related algorithms will be briefly described.

### 3. ALGORITHMS

Algorithms for finding optimal solutions can be classified into adaptive and non-adaptive methods [11]. Non-adaptive methods first determine all search points at which the function is to be evaluated (i.e. points on a grid) and then evaluate the objective function at all these sites and determine the approximation of the optimal solution based on the results. Design of Experiments (DoE) and Random-Sampling (also known as Trial and Error) belong to this category. Adaptive methods, such as direct and evolutionary search methods take the results of previous evaluations into account when determining a new search point. Surprisingly, for very general classes of continuous functions, adaptive methods do not have a better worst-case performance than non-adaptive methods [11]. However, there is some evidence that the average case complexity of adaptive methods is better. However, in practice it can be misleading to work with an idealized assumption about the function geometry, and it is best practice to measure the performance of algorithms on representative examples from the problem domain. Design of experiments is a method that is often not used for optimization in the first place. Rather it is

used to study the effect of input variables (or combinations of them) of the system on their output variables. In our study we will use full factorial design which allows the effect of all input variables and combinations of them to be studied. As a second non-adaptive method we will apply pure random search, i.e. generating uniformly random parameter vectors within the bounds and evaluating them. This strategy will be used to check the problem difficulty and the added value of the more sophisticated adaptive optimization algorithms.

As representatives of different categories of adaptive search methods, direct search, evolution strategies, and gradient-based strategies will be tried.

A characteristic of *direct search* methods is that they can deal with non-differentiable problems. A direct search method neither computes nor approximates derivatives. In contrast to evolution strategies they schedule experiments deterministically, using heuristics to search for improvements in the vicinity of the current best solutions, such as placing new candidate solutions on adjacent grid points of a multi-dimensional grid spanned over the search space and comparing the current function value with the best obtained value thus far. If the current value is better it will be the new best value and the algorithm moves to this point. If no better function value can be found, the grid density is increased to reduce the step sizes that are taken. The direct search method that is used for optimization in this study is the Hooke and Jeeves pattern search [4]. This algorithm is a representative of a class of algorithms that today are classified under the framework of generalized pattern search. Convergence to a stationary point (a local optimum or a saddle point) was proven recently by Lewis and Torczon [7].

Evolution strategies (ES) (see for example [1]) are optimization algorithms which use principles of biological evolution in a simulated way to find optima of an objective function. The idea is to apply mutation (random Gaussian perturbation of the search point) and selection alternating on a population or single solution in order to gradually improve its function value. Evolution strategies feature a control of the perturbation strength based on the success rate. They are considered to be robust search algorithms that can be used for non-differentiable function optimization.

The *Gradient descent method* searches in the direction of the negative gradient of the function at the current point to find the minimum. If a derivative is not available for the objective function, it can be computed numerically by means of finite differences. This can be time consuming if the objective function computation is time consuming and the dimensionality is high. For each dimension two objective function calls are necessary to provide approximated gradients. To take advantage of the gradient a line search is used along the negative gradient direction. When no further improvement can be made, the algorithm computes a new gradient vector at the relative optimum obtained. With some adjustments a gradient descent algorithm can be transformed into a Quasi Newton method, for

example the BFGS algorithm. In BFGS a Hessian matrix is built up during the optimization. BFGS avoids the zigzag behavior that sometimes occurs in the gradient descent method due to the fact that the gradient direction is perpendicular to the old gradient direction at the relative minimum. A description of both algorithms can be found in Press et al. [9]. The discussion of a multi-criterion search algorithm we will postpone to Section 5.

### 3. DESIGN OF EXPERIMENTS AND RANDOM SAMPLING

Design of Experiments is a useful technique to analyze the effect of parameters and combinations of parameters (interaction) on the objective function value. The technique used here is called full factorial design. This technique is applied as follows: We assume that for each variable an upper and lower limit is given and we want to measure the effect of the variable within its range. Given these ranges, the search space forms an  $N$ -dimensional hypercube, with  $N$  being the number of input variables. Full factorial design evaluates the objective function at all  $2^N$  corners of the hypercube (see Figure 2). The results are used to fit a multi-linear model the coefficients of which can be interpreted as effects and interactions, respectively.

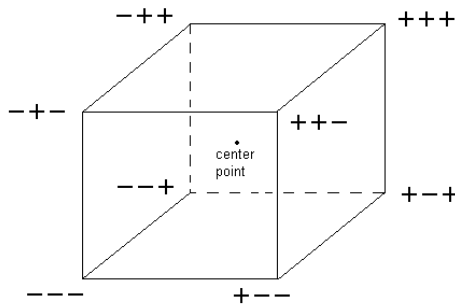


Figure 2. Placement of experiments in a box constrained search space using full factorial design.

In the given case of four variables one obtains the following multilinear form:

$$y = a_0 + a_1x_1 + a_2x_2 + a_3x_3 + a_4x_4 + a_{12}x_1x_2 + a_{13}x_1x_3 + a_{14}x_1x_4 + a_{23}x_2x_3 + a_{24}x_2x_4 + a_{34}x_3x_4 + a_{123}x_1x_2x_3 + a_{124}x_1x_2x_4 + a_{134}x_1x_3x_4 + a_{234}x_2x_3x_4 + a_{1234}x_1x_2x_3x_4$$

This form contains  $2^N$  coefficients that can be determined by fitting a polynomial approximation. Setting the level of the lower bound to 0 and the level of the upper bound to 1, one obtains:

$$y = 21376.12 - 530.24x_1 + 5075.46x_2 + 3427.96x_3 + 4040.30x_4 + 62.71x_1x_2 - 232.61x_1x_3 - 279.24x_1x_4 - 203.37x_2x_3 - 249.24x_2x_4 + 904.03x_3x_4 + 19.69x_1x_2x_3 + 22.35x_1x_2x_4 + 58.92x_1x_3x_4 - 14.43x_2x_3x_4 - 33.37x_1x_2x_3x_4$$

The result can be interpreted in the following way: Parameter  $x_2$  (window fraction) has an important role in the objective function value as already shown in the scatter plots. Parameters  $x_3$  (load equipment) and  $x_4$  (load lighting) also evidently contribute to the objective function value. Parameters  $x_3$  and  $x_4$  in combination with  $x_1$  or  $x_2$  have a decreasing effect on the objective function value. As opposed to factorial DoE, a space filling design distributes search points uniformly in the search space. The simplest strategy is random sampling (or Trial and Error). With random sampling we measured the objective function value at a set of 250 points. The results are visualized in a Parallel Coordinates Diagram (see figure 4), where each line corresponds to one evaluated solution. The range of objective function values (energy consumption) is wide (ca. 9200K-33000K). The best objective function values are obtained for relatively small values of all parameters. This accords with results of the factorial DoE.

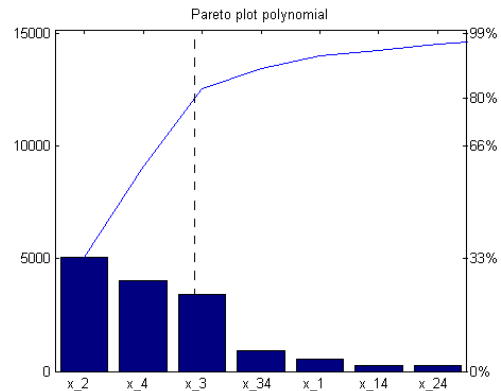


Figure 3. Pareto Plot of Results obtained with experimental design. The dashed line marks the part where 80% of the effect has been accumulated.

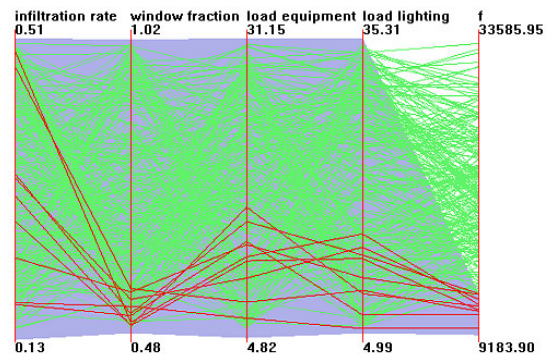


Figure 4. Results of a DoE visualized with a Parallel Coordinates Diagram (XMVD).

The Pareto Plot in Figure 3 summarizes the results obtained and it clearly shows that based on the linear approximation the parameters  $x_2$ ,  $x_4$ , and  $x_3$  control the energy consumption, while  $x_1$  has a minor effect on it.

## 6. ADAPTIVE ALGORITHMS

Next, we studied the question of whether adaptive optimization can help to further reduce the objective function value, and tested different optimization strategies for this purpose. For each run of the algorithms the maximum allowed number of objective function calls to evaluate is 100. The choice of a relatively small budget of function evaluations was motivated by the high time-consumption of the simulator-based evaluation. It became clear after the first couple of runs that both parameters, load equipment and load lighting, contribute to a low energy consumption if these parameters are as low as possible. To get more interesting information from this research the parameter range was modified one time. Algorithms were run for each problem nine times (nine different seeds/starting points). By running the same algorithm more than once single good or bad runs are smoothed out.

For the different types of algorithms abbreviations are used in the figures below. Graphs “1p1”, “4p20”, and “3p10” are evolution strategies, “pattern” (search) is the Hooke Jeeves pattern search method, and “gradient” stands for the gradient-based BFGS method. The (1+1) evolution strategy uses a  $1/5^{\text{th}}$  success rule for the self-adaptation of the mutation step-size. 4p20 (3p10) is an evolution strategy using a parent population of size 4 (3) and an offspring population of size 20 (10). The mutation-based self-adaptation of step sizes in these strategies conforms to that described by [1]. Because of the small number of function evaluations we did not apply covariance matrix adaptation in the evolution strategy. Initialization is uniform randomly within the bounds. Intervals are treated by setting the parameter value to that of the boundary, if it exceeds it after mutation. Hooke and Jeeves search, gradient-based methods, and BFGS were implemented as suggested in *Numerical Recipes in C* [9].

The graphs in figures 5 and 6 show the course of the objective function values during the optimization in relation to the current evaluation. The evolution strategies have many more peaks caused by the random effects in these algorithms. The pattern search and gradient descent algorithms follow for this problem an almost monotonous path to a minimum, which is obtained after only 15 and 30 iterations, respectively.

We assume that for more complex problem with a higher dimension evolution strategies may be more competitive. Moreover, similar methods will be considered in Section 5 for finding a Pareto front of a multi-criterion problem. In this case the fact that evolutionary algorithms can work with a set of solutions rather than a single solution becomes crucial.

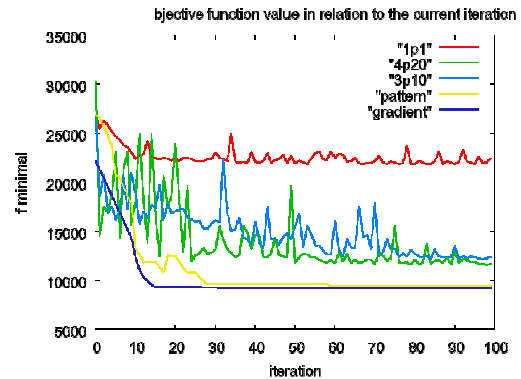


Figure 5. Course of objective function; load equipment 6-30W/m<sup>2</sup>

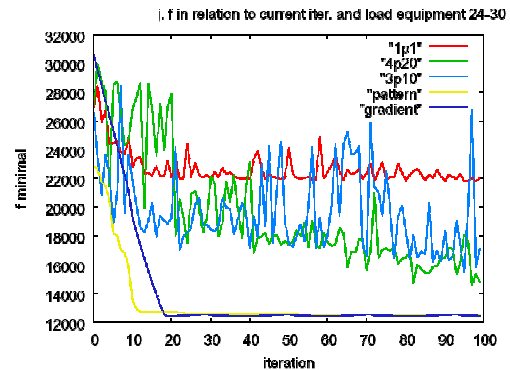


Figure 6. Course of objective function; Load equipment 24-30W/m<sup>2</sup>

An overview of the statistics obtained from the different runs of the algorithms can be found in table 1. Notable is the fact that both pattern and gradient search perform well for almost all the runs between every possible boundary for the load equipment parameter. Both methods have one diverged run for the case that parameter load equipment has boundaries between 6 and 30. The gradient descent algorithm shows the most reliable behavior. The standard deviation is low or close to zero. The 1p1 algorithm performs consistently badly. Both 3p10 and 4p20 come up with at least one good result out of the nine runs, but there is a big difference between the best and the worst runs. The reliability of these two algorithms is not very high. Possibly, the location of the optimum on the boundary and the low budget of evaluations have a negative effect on the competitiveness of ES with deterministic techniques.

Name	Load equipment	Minimum	Median	Maximum	Mean	Standard deviation
1p1	6-12	15212,168	15395,21	16391,93	15746,43	482,11
4p20	6-12	9590,21	10458,95	12292,57	10757,04	915,58
3p10	6-12	9413,72	12656,96	14070,37	12181,51	1523,42
pattern	6-12	9276,13	9346,4	9681,82	9415,96	146,72
gradient	6-12	9243,79	9243,79	9243,86	9243,8	0,02
1p1	6-30	21494,9	21886,34	22573,75	22068,61	433,40
4p20	6-30	9457,91	11604,6	14959,55	11683,23	1753,53
3p10	6-30	9468,22	12160,26	17004,12	12491,29	2405,65
Pattern	6-30	9347,63	9462,74	31697,94	11966,90	6977,85
Gradient	6-30	9243,8	9244,16	12746,16	9633,19	1100,60

Table 1. Statistics of the runs on two different problems

An interesting question that arises is whether the results of adaptive optimization are better than that those achieved with non-adaptive random sampling. If the objective values of the red lines in figure 4 are compared to the results of the algorithms the 1p1 evolution strategy performs worse than uniformly sampling the search space. However, it should be noted that the run of an algorithm takes at most 100 calls of the objective function while the random set is built with 250 calls to the objective function. The 3p10 and 4p20 evolution strategies on the other hand perform better than uniform sampling. The minimum values found with these algorithms are lower than that of the best sample. The medians of both algorithms measured over nine runs, however, are worse than the best random sample. Both, pattern and gradient search have a minimum objective function value and median which are better than the best value found by random search. These two adaptive algorithms can thus be highly recommended for this problem, in the case of single objective optimization.

## 6. PARETO OPTIMIZATION

A main goal for future integration will be the handling of multiple objectives. In the case of two or three objectives, methods for approximating Pareto fronts seem to be a promising methodology [12]. They determine (an approximation to) the set of efficient solutions, which is the subset of the search points; these are not dominated by other solutions in the search space with respect to Pareto dominance. After this set has been determined the expert can select solutions based on the visualization of the Pareto front, showing the trade-offs and attainable solutions qualities. For computing a small, well-distributed, set of points on the Pareto front the SMS-EMOA [3] is an advanced EMOA and it outperforms other, more popular techniques, such as NSGA-II and SPEA-II, on the standard benchmark problems and performance metrics (ZDT and DTLZ). A description and extensive benchmark results can be found in Beume et al. [2]. A particular advantage of this method in the context of building optimization is that it is well suited for approximation of the Pareto front with a small number of approximation points. This makes it also very appropriate for optimization with a limited budget of

function evaluations. The basic idea of the SMS-EMOA is to repetitively generate small random variations of points in the existing approximation set, and discard solutions from the set based on dominance ranking and the contributions of non-dominated points to the dominated area of the approximation set.

The two objectives are: firstly, minimization of the cooling hours exceeding 28°C. This resembles closely the Dutch criterion for thermal comfort related to the overheating hours in the building; secondly, minimization of the energy demands of the building.

The Pareto front approximation (figure 7) was computed with 1000 function evaluations and a set of 10 approximation points. Figure 8 reveals that the randomized algorithm reliably finds sets close to this solution. We also compared the algorithm to the NSGA-II algorithm, a popular method for Pareto optimization and obtained similar results. The NSGA-II seems to have less stable convergence behaviour on this optimization problem (Figure 9), while the median attainment surfaces of the two algorithms look very similar (Figure 10).

The set of points is relatively small but in our opinion sufficient to get an impression of the shape of the Pareto front and select interesting points. Increasing the size of the set will cause a slow down of convergence time which cannot be afforded, due to the small budget of objective function evaluations.

The Pareto front is convex and has a linear trade-off between the two criteria in the range of an energy consumption of 1000-1500 [kWh] and 5400-6500 hours exceeding 28°C per year. In the range 850-1000 [kWh] the value of the hours exceeding 28°C grows progressively with savings in energy consumption.

It can be observed that the amount of energy consumption varies within the boundaries advised by the BESTTEST [6]. However, the number of hours above 28°C exceeds the comfort Dutch comfort criterion, weighted overheating hours, which is caused by using a simplified measure, air temperature. As mentioned earlier, the simplification does not spoil the approach to assess the different algorithms determined in the outcomes.

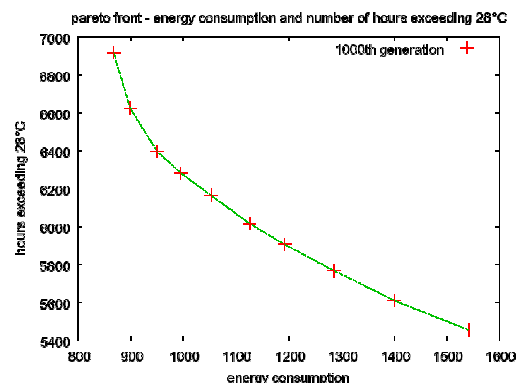


Figure 7. Approximation of the Pareto front of the BESTTEST problem obtained with the SMS-EMOA.

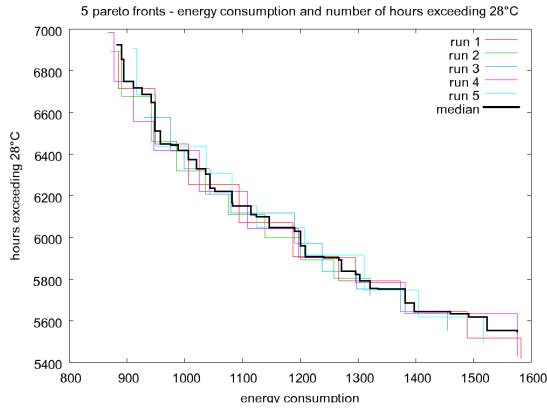


Figure 8. Summary of five results of five different runs with the SMS EMOA. The black line depicts the average attainment surface,

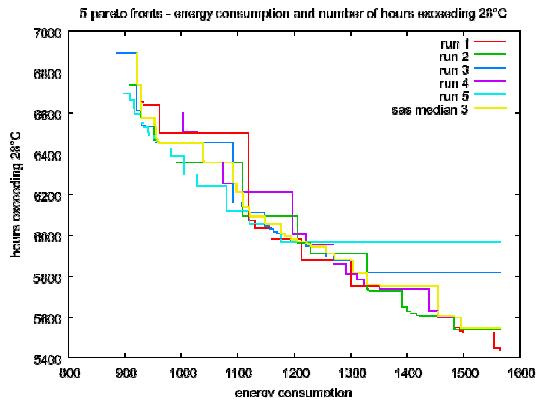


Figure 9. Summary of five results of five different runs with the NSGA-II. The yellow line depicts the average attainment surface.

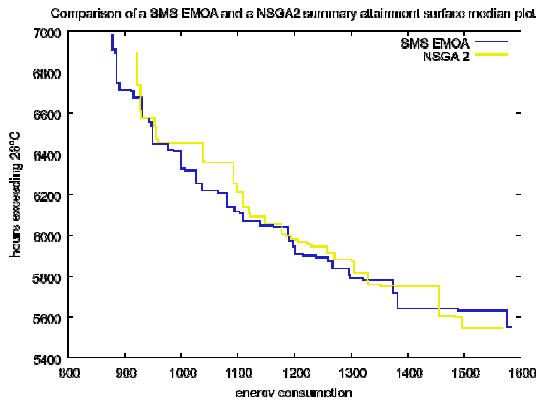


Figure 10. Comparison of the median attainment curve of NSGA-II and SMS EMOA.

## 6. SUMMARY

We presented first results and experiences of our endeavor to build optimization and design space exploration tools for future integration in a Building Performance Tool. DoE and Random Sampling

techniques turn out to be useful for exploration but fail to provide optimal settings of the system parameters in this domain. The results for single objective optimization of the BESTTEST case problem indicate that adaptive, deterministic techniques are favorable for this problem domain. Though in almost all cases random search was outperformed, results with the tested evolutionary and deterministic strategies differ strongly, with an advantage for deterministic direct search and gradient based methods. As a more general message, the study shows thus the importance of considering a spectrum of different techniques rather than a single method for optimization when introducing optimization to a new problem domain.

Another problem tackled was the optimization of conflicting objectives (here thermal comfort versus energy consumption). The proposed technique is Pareto optimization with small population sizes. An interesting result was obtained using the SMS-EMOA, an advanced population-based evolutionary algorithm. An evenly distributed set of points was obtained that visualizes the almost linear trade-off between thermal comfort and energy consumption within the set of efficient solutions.

Future work will have to deal with a more extensive study using more representative examples from the domain. An interesting question will be, whether in high-dimensional problems evolution strategies are more competitive than in the given low-dimensional case. Furthermore the number of algorithms tested is still comparatively low and various parameter settings, deviating from default settings, may be analyzed. To do this efficiently automatic parameter tuning using design of experiments seems to be a promising technique. For the ES de-randomized step-size adaptation and alternative interval boundary treatment methods may enhance their performance.

In addition we intend to deepen the study in Pareto optimization, comparing multi-start method to population-based methods and refine the choice of optimization criteria. In this study canonical algorithms were applied and it could be interesting to develop tailored optimization algorithms for the problem domain. However, the study at hand proves that single- and multi-objective adaptive optimization is an interesting approach as an alternative to trial-and-error strategies in the domain of building performance design simulation and it can also be applied when the budget of function evaluation is very limited.

## REFERENCES

- [1] Beyer H.-G., Schwefel H.-P. (2002): *Evolution strategies - A comprehensive introduction*. Natural Computing 1 (1); pp 3-52.
- [2] Beume, N, B. Naujoks, und M. Emmerich. (2007): *SMS-EMOA: Multiobjective selection based on dominated hypervolume*. European Journal of Operational Research, 181(3):1653-1669, 2007.

- [3] Emmerich, M., Beume N., and Naujoks B.: *An EMO Algorithm using the hypervolume measure as a selection criterion*, EMO 2005, LNCS 3410, Springer, pp. 62-76
- [4] Hooke, R. and T.A. Jeeves: *Direct search solution to numerical and statistical problems*, J. Assoc. Comput. Mach. 8 (1961), pp. 212-229
- [5] Hopfe, C.J., Struck, C., Hensen J., Boehms, M (2006): *Adapting engineering design approaches to building design- potential benefits*, proceedings of 6<sup>th</sup> Int. Postgraduate Research Conf. in the Built and Human Environment, 6-7 April, Technische Universiteit Delft, BuHu, University of Salford, p.369-378 5, 1989
- [6] Judkoff, R. and Neymark, J. 1995. *International energy agency building energy simulation test (BESTEST) and diagnostic method*, National Renewable Energy Laboratory, Golden, CO.
- [7] R.M. Lewis and V. Torczon: *Pattern search algorithm for bound constrained optimization*, SIAM J. Optim. 9 (1999), pp.1082-1099
- [8] Michalek, J.J., Choudhary R., Papalambros P.Y. (2002): *Architectural Layout Design Optimization*
- [9] Press W., S. Teukolsky, Vetterling P., Flannery B. (1992) *Numerical Recipes in C: The Art of Scientific Computing (2nd edition)*, Cambridge University Press.
- [10] Parmee, I and Hajela, P.: *Optimization in Industry*, Springer, 2002.
- [11] Ritter, K. and P. Novak (1996): *Global optimization using hyperbolic cross points*, In: State of the Art in Global Optimization (C. A. Floudas, P. M. Pardalos, eds.), pp. 19-33, Kluwer, Dordrecht.
- [12] Siarry, P. and Y. Collette: *Multiobjective Optimization: Principles and Case Studies*, Springer. 2003
- [13] Wilde, Pieter de, 2004, "Computational Support for the Selection of Energy saving building components", PhD-thesis, Delft Univ. of Technology, Faculty of Architecture, Building Physics Group, Delft, The Netherlands
- [14] Wright, Jonathan, Zhang, Y., Angelov, P.P., Buswell, R.A. and Hanby, V.I. (2004): *Building system design synthesis and optimization*, pp., Final Report to ASHRAE on Research Project 1049-RP.